



EUON Workshop, 25<sup>th</sup> September 2014

# **OOPS! (ONTOLOGY PITFALL SCANNER!): A WEB-BASED TOOL FOR ONTOLOGY EVALUATION**

**María Poveda-Villalón**, Mari Carmen Suárez-Figueroa and  
Asunción Gómez-Pérez

Ontology Engineering Group.

Universidad Politécnica de Madrid.

{**mpoveda**, mcsuarez, asun}@fi.upm.es

OOPS! website: <http://www.oeg-upm.net/oops>

Twitter account: @OOPSoeg



- **Introduction**
- Pitfall Catalogue
- OOPS! (OntOlogy Pitfall Scanner!)
- Conclusions and Future Work

**Methodologies** (e.g: Methontology [1, 2], On-To-Knowledge [3], DILIGENT [4], and the NeOn Methodology [5]) that support the **ontology development** transformed the art of building ontologies into an **engineering activity**.

The correct application of such **methodologies** benefits the **ontology quality**.

**However**

Developers must tackle a wide range of **difficulties** and **handicaps** when **modelling ontologies**.



These difficulties can imply the appearance of **anomalies** or **worst practices** in ontologies.



**Ontology evaluation** (checking the technical quality of an ontology against a frame of reference) is a **crucial activity** in ontology engineering projects.



- [1] Gómez-Pérez, A., Fernández-López, M., Corcho, O. Ontological Engineering. November 2003. Springer Verlag. Advanced Information and Knowledge Processing series. ISBN 1-85233-551-3.
- [2] M. Fernández-López, A. Gómez-Pérez, N. Juristo. METHONTOLOGY: From Ontological Art Towards Ontological Engineering. 1997. Spring Symposium on Ontological Engineering of AAAI. Stanford University, California, pp 33–40.
- [3] S. Staab, H.P. Schnurr, R. Studer, Y. Sure. Knowledge Processes and Ontologies. IEEE Intelligent Systems 16(1):26–34. (2001).
- [4] H. S. Pinto, C. Tempich, S. Staab. *DILIGENT: Towards a fine-grained methodology for Distributed, Loosely-controlled and evolving Engineering of oNTologies*. In Ramón López de Mantaras and Lorenza Saitta, Proceedings of the 16th European Conference on Artificial Intelligence (ECAI 2004), August 22nd - 27th, pp. 393–397. IOS Press, Valencia, Spain, August 2004. ISBN: 1-58603-452-9. ISSN: 0922-6389.
- [5] M.C. Suárez-Figueroa. Doctoral Thesis: NeOn Methodology for Building Ontology Networks: Specification, Scheduling and Reuse. Spain. Universidad Politécnica de Madrid. June 2010.



A lot of work has been done in ontology evaluation:









- generic quality evaluation frameworks [1, 2, 3, 4, 5],
- methods based on the final (re)use of the ontology [6]
- quality models based on features, criteria and metrics [7, 8]
- tools: ODEclean, ODEval, XDTools, OntoCheck, EyeBall, MoKi, etc.

However

Ontology evaluation is still largely neglected by developers and practitioners

Maybe because...

- (a) the current **time-consuming** and **tedious** nature of evaluating the quality of an ontology
- (b) the **lack of awareness** of the necessity for evaluating ontologies we are producing and publishing throughout the web.

- 
- [1] Welty, C.A., and Guarino, N. Supporting ontological analysis of taxonomic relationships. In Data & Knowledge Engineering. vol 39, pp 51-74. 2001
- 
- [2] Duque-Ramos, A., Uriel López, J. T. Fernández-Breis, Robert Stevens. *Towards an SQUaRE-based Quality Evaluation Framework for Ontologies*. OntoQual 2010 - Workshop on Ontology Quality at EKAW 2010) ISBN: ISSN 1613-0073. Pages: 13-24. 15 October 2010. Lisbon, Portugal.
- 
- [3] Gangemi, A., Catenacci, C., Ciaramita, M., Lehmann J. *Modelling Ontology Evaluation and Validation*. Proceedings of ESWC2006, number 4011 in LNCS, Budva. 2006.
- 
- [4] Gómez-Pérez, A. *Ontology Evaluation*. Handbook on Ontologies. S. Staab and R. Studer Editors. Springer. International Handbooks on Information Systems. Pp: 251-274. 2004.
- 
- [5] Strassunskas, D., Tomassen, S.L.: *The role of ontology in enhancing semantic searches: the EvOQS framework and its initial validation*. Int. J. Knowledge and Learning, Vol. 4, No. 4, pp. 398-414.
- 
- [6] Suárez-Figueroa, M.C. Doctoral Thesis: NeOn Methodology for Building Ontology Networks: Specification, Scheduling and Reuse. Spain. Universidad Politécnica de Madrid. June 2010.
- 
- [7] Flemming, A.. Assessing the quality of a Linked Data source. Proposal. <http://www2.informatik.hu-berlin.de/~flemming/Proposal.pdf>
- 
- [8] Burton-Jones, A., Storey, V.C., and Sugumaran, V., and Ahluwalia, P. A Semiotic Metrics Suite for Assessing the Quality of Ontologies. Data and Knowledge Engineering, (55:1) 2005, pp. 84-102.

## Our objective is

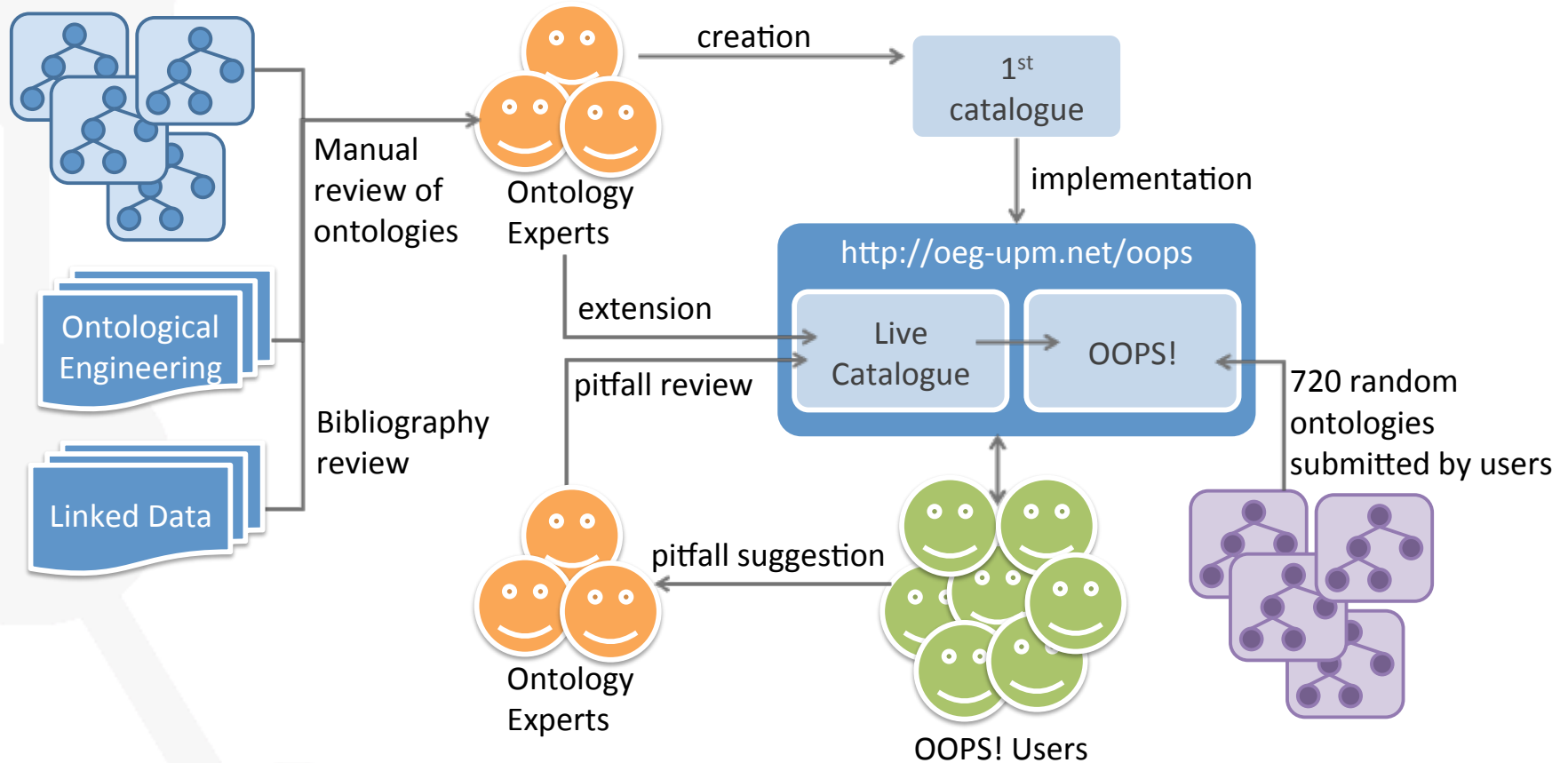
- To **ease** the activity of **ontology evaluation** (mainly people who are not ontological engineers)
- To **reduce time** and **effort** in **ontology evaluation**

## For doing so we have

- (a) **Created** a **catalogue** contained potential errors we have seen in other ontologies and other authors' work
- (b) Established mechanisms to keep this **catalogue updated** and **maintained**
- (c) **Automated** the detection of several errors
- (d) **Provided** a web-based **user interface** and a **Restful web service**

- Introduction
- **Pitfall Catalogue**
- OOPS! (OntOlogy Pitfall Scanner!)
- Conclusions and Future Work

# Creation of the pitfall catalogue & maintenance process

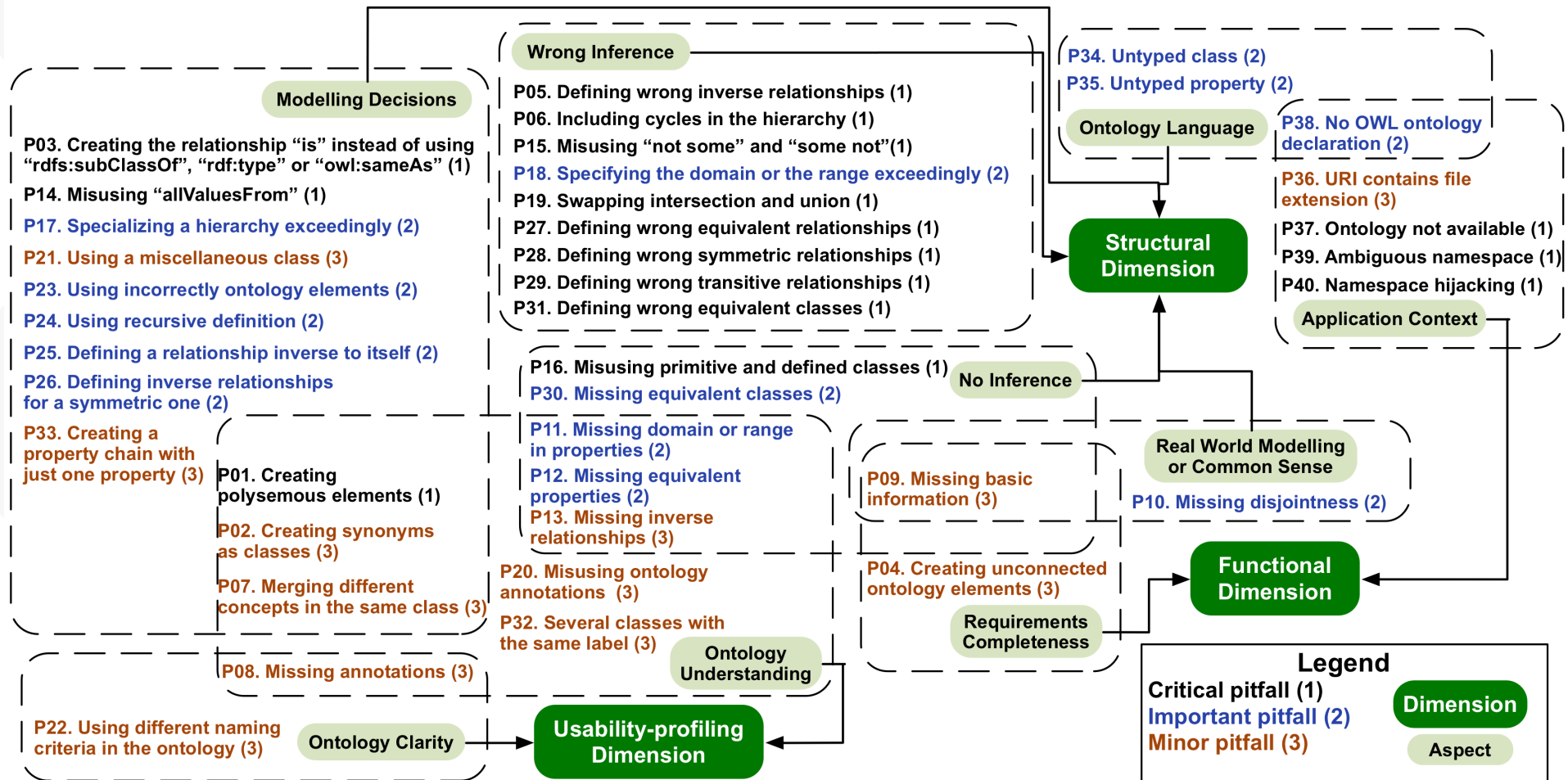


- Pitfalls **could represent** or **lead to** an error.
- Pitfalls are **not necessarily** errors. For example, pitfalls **might not** represent an error depending on:
  - Modelling decisions.
  - Context or scope of the ontology.
  - Ontology requirements.
- In addition not all the pitfalls are equally important.
- An **indicator (critical, important, minor)** has been **associated** to each pitfall according to their **possible** negative consequences
  - **Survey** on ontology pitfalls importance: <http://goo.gl/SEddMN>



# Pitfall Catalogue so far

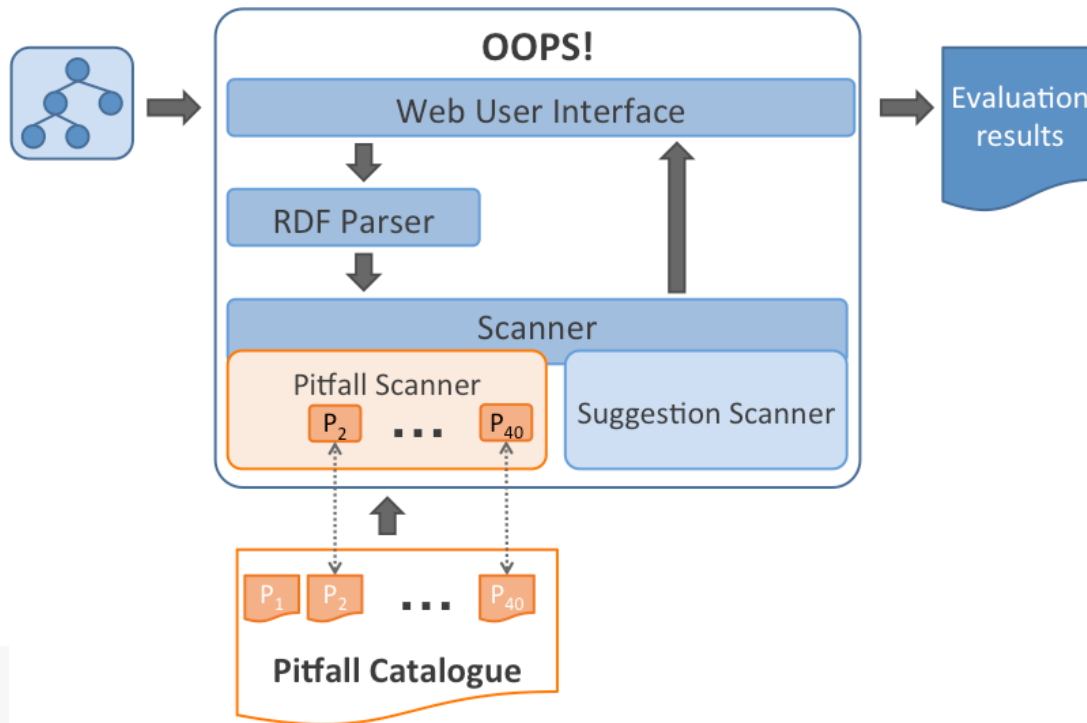
[www.oeg-upm.net/oops/catalogue.jsp](http://www.oeg-upm.net/oops/catalogue.jsp)



- Introduction
- Pitfall Catalogue
- **OOPS! (OntOlogy Pitfall Scanner!)**
- Conclusions and Future Work

# OOPS! - How it is internally organized (i)

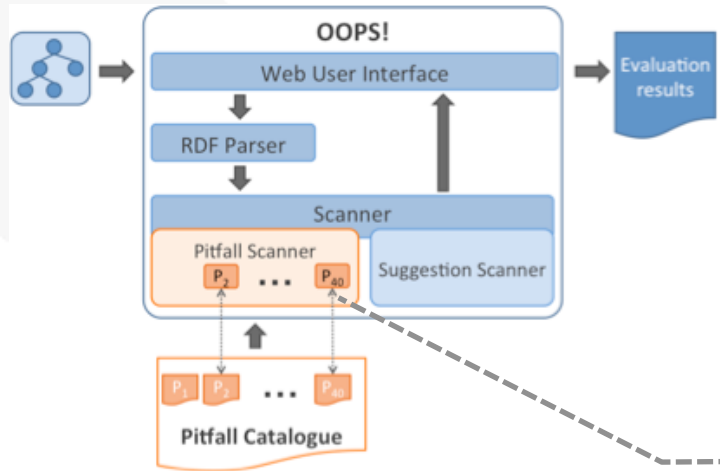
- **Web-based** tool
- Available at <http://www.oeg-upm.net/oops>
- Ontology development environment **independent**
- **No installation** process required



**Jena API:** <http://jena.sourceforge.net/>  
**Java EE:** <http://www.oracle.com/technetwork/java/javaee/overview/index.html>  
**HTML:** <http://www.w3.org/html/wg/>

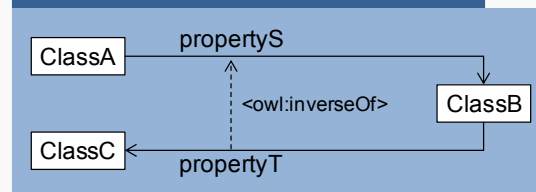
**jQuery:** <http://jquery.com/>  
**JSP:** <http://www.oracle.com/technetwork/java/javaee/jsp/index.html>  
**CSS:** <http://www.w3.org/Style/CSS/>

# OOPS! - How it is internally organized (ii)



- **32 pitfalls implemented** out of 40 included in the catalogue
- 1 Java class per pitfall implementation
- **Detection** automated in 3 ways:
  - **Lexical content analysis:** make use of the content of annotations and identifiers for detecting pitfalls. E.g: P22: Using different naming criteria in the ontology.
  - **Seeking a particular characteristic:** check general characteristics of the ontology not related to the internal structure of the ontology or to the content of the lexical entities. E.g: P36. URI contains file extension.
  - **Structural pattern:** analyze the internal structure of the ontology, seeking specific parts of the model . E.g: P5: Defining wrong inverse relationships.

## P5. Defining wrong inverse relationships





## OntOlogy Pitfall Scanner!

**OOPS! (OntOlogy Pitfall Scanner!)** helps you to detect some of the most common pitfalls appearing when developing ontologies.

To try it, enter a URI or paste an OWL document into the text field above. A list of pitfalls and the elements of your ontology where they appear will be displayed.

Scanner by URI:

Scanner by URI

Example: [http://data.semanticweb.org/ns/swc/swc\\_2009-05-09.rdf](http://data.semanticweb.org/ns/swc/swc_2009-05-09.rdf)

Scanner by direct input:

If you just include the RDF code here, the following Pitfalls will not be checked: P36. URI contains file extension, P37. Ontology not available, P40. Namespace hijacking

Scanner by RDF

☒ Uncheck this checkbox if you don't want us to keep a copy of your ontology.

[Go to advanced evaluation](#)

## News!

Now you can **integrate OOPS!** pitfall detection **with your own developments and tools** simply by invoking the **OOPS! RESTful Web Service**.

## Detecting common pitfalls in ontologies

Modelling ontologies has become one of the main topics of research within ontological engineering because of the difficulties it involves. Developers must tackle a wide range of difficulties and handicaps when modelling ontologies that can imply the appearance of anomalies or errors in ontologies. Therefore, it is important to evaluate the ontologies in order to detect those potential problems.

In this sense, OOPS! helps you to detect some of the most common pitfalls appearing within ontology developments. For example, OOPS! warns you when:

- The domain or range of a relationship is defined as the intersection of two or more classes. This warning could avoid reasoning problems in case those classes could not share instances.
- No naming convention is used in the identifiers of the ontology elements. In this case the maintainability, the accessibility and the clarity of the ontology could be improved.
- A cycle between two classes in the hierarchy is included in the ontology. Detecting this situation could avoid modelling and reasoning problems.
- And many other problems described in [the catalogue](#).

Have a look at OOPS! results for the [DBpedia 3.8](#) and [AKT Reference Ontology \(Portal Ontology\)](#) ontologies.

Please, help us making OOPS! better. **Feedback** is more than welcome and you can also **suggest new pitfalls!**

Want to help?

- [Suggest new pitfalls](#)
- [Provide feedback](#)

Documentation:

- [Pitfall catalogue](#)
- [User guide](#)
- [Technical report](#)


Related papers:

- [EKAW 2012](#)
- [ESWC 2012 Demo](#)
- [Ontoqual 2010](#)
- [CAEPIA 2009](#)

Web services:

- [RESTful Web Service](#)

Ontology  
input area

 **OntOlogy Pitfall Scanner!**

**OOPS! (OntOlogy Pitfall Scanner!)** helps you to detect some of the most common pitfalls appearing when developing ontologies.  
To try it, enter a URI or paste an OWL document into the text field above. A list of pitfalls and the elements of your ontology where they appear will be displayed.

Scanner by URI:

Example: [http://data.semanticweb.org/ns/swc/swc\\_2009-05-09.rdf](http://data.semanticweb.org/ns/swc/swc_2009-05-09.rdf)

Scanner by direct input:

☒ Uncheck this checkbox if you don't want us to keep a copy of your ontology. [Go to advanced evaluation](#)

If you just include the RDF code here, the following Pitfalls will not be checked: P36. URI contains file extension, P37. Ontology not available, P40. Namespace hijacking

## News!

Now you can **integrate OOPS!** pitfall detection **with your own developments and tools** simply by invoking the [OOPS! RESTful Web Service](#).

## Detecting common pitfalls in ontologies

Modelling ontologies has become one of the main topics of research within ontological engineering because of the difficulties it involves. Developers must tackle a wide range of difficulties and handicaps when modelling ontologies that can imply the appearance of anomalies or errors in ontologies. Therefore, it is important to evaluate the ontologies in order to detect those potential problems.

In this sense, OOPS! helps you to detect some of the most common pitfalls appearing within ontology developments. For example, OOPS! warns you when:

- The domain or range of a relationship is defined as the intersection of two or more classes. This warning could avoid reasoning problems in case those classes could not share instances.
- No naming convention is used in the identifiers of the ontology elements. In this case the maintainability, the accessibility and the clarity of the ontology could be improved.
- A cycle between two classes in the hierarchy is included in the ontology. Detecting this situation could avoid modelling and reasoning problems.
- And many other problems described in [the catalogue](#).

Have a look at OOPS! results for the [DBpedia 3.8](#) and [AKT Reference Ontology \(Portal Ontology\)](#) ontologies.

Please, help us making OOPS! better. **Feedback** is more than welcome and you can also [suggest new pitfalls!](#)

### Want to help?

- [Suggest new pitfalls](#)
- [Provide feedback](#)

### Documentation:

- [Pitfall catalogue](#)
- [User guide](#)
- [Technical report](#)

### Related papers:

- [EKAW 2012](#)
- [ESWC 2012 Demo](#)
- [Ontoqual 2010](#)
- [CAEPIA 2009](#)

### Web services:

- [RESTful Web Service](#)



**OOPS! (Ontology Pitfall Scanner!)** helps you to detect some of the most common pitfalls appearing when developing ontologies. To try it, enter a URI or paste an OWL document into the text field above. A list of pitfalls and the elements of your ontology where they appear will be displayed.

Scanner by URI:

Example: [http://data.semanticweb.org/ns/swc/swc\\_2009-05-09.rdf](http://data.semanticweb.org/ns/swc/swc_2009-05-09.rdf)

Scanner by direct input:

If you just include the RDF code here, the following Pitfalls will not be checked: P36. URI contains file extension, P37. Ontology not available, P40. Namespace hijacking

☒ Uncheck this checkbox if you don't want us to keep a copy of your ontology.

[Go to advanced evaluation](#)

## News!

Now you can **integrate OOPS!** pitfall detection **with your own developments and tools** simply by invoking the [OOPS! RESTful Web Service](#).

## Detecting common pitfalls in ontologies

Modelling ontologies has become one of the main topics of research within ontological engineering because of the difficulties it involves. Developers must tackle a wide range of difficulties and handicaps when modelling ontologies that can imply the appearance of anomalies or errors in ontologies. Therefore, it is important to evaluate the ontologies in order to detect those potential problems.

In this sense, OOPS! helps you to detect some of the most common pitfalls appearing within ontology developments. For example, OOPS! warns you when:

- The domain or range of a relationship is defined as the intersection of two or more classes. This warning could avoid reasoning problems in case those classes could not share instances.
- No naming convention is used in the identifiers of the ontology elements. In this case the maintainability, the accessibility and the clarity of the ontology could be improve.
- A cycle between two classes in the hierarchy is included in the ontology. Detecting this situation could avoid modelling and reasoning problems.
- And many other problems described in [the catalogue](#).

Have a look at OOPS! results for the [DBpedia 3.8](#) and [AKT Reference Ontology \(Portal Ontology\)](#) ontologies.

Please, help us making OOPS! better. **Feedback** is more than welcome and you can also **suggest new pitfalls!**

Want to help?

- [Suggest new pitfalls](#)
- [Provide feedback](#)

Documentation:

- [Pitfall catalogue](#)
- [User guide](#)
- [Technical report](#)

Related papers:

- [EKAW 2012](#)
- [ESWC 2012 Demo](#)
- [Ontoqual 2010](#)
- [CAEPIA 2009](#)

Web services:

- [RESTful Web Service](#)

Ontology input area

Brief description

# OOPS! - How it works (i)



**OOPS! (Ontology Pitfall Scanner!)** helps you to detect some of the most common pitfalls appearing when developing ontologies.

To try it, enter a URI or paste an OWL document into the text field above. A list of pitfalls and the elements of your ontology where they appear will be displayed.

Scanner by URI:

Example: [http://data.semanticweb.org/ns/swc/swc\\_2009-05-09.rdf](http://data.semanticweb.org/ns/swc/swc_2009-05-09.rdf)

Scanner by URI

If you just include the RDF code here, the following Pitfalls will not be checked: P36. URI contains file extension, P37. Ontology not available, P40. Namespace hijacking

Scanner by direct input:

Scanner by RDF

☒ Uncheck this checkbox if you don't want us to keep a copy of your ontology.

[Go to advanced evaluation](#)

## News!

Now you can **integrate OOPS!** pitfall detection **with your own developments and tools** simply by invoking the [OOPS! RESTful Web Service](#).

## Detecting common pitfalls in ontologies

Modelling ontologies has become one of the main topics of research within ontological engineering because of the difficulties it involves. Developers must tackle a wide range of difficulties and handicaps when modelling ontologies that can imply the appearance of anomalies or errors in ontologies. Therefore, it is important to evaluate the ontologies in order to detect those potential problems.

In this sense, OOPS! helps you to detect some of the most common pitfalls appearing within ontology developments. For example, OOPS! warns you when:

- The domain or range of a relationship is defined as the intersection of two or more classes. This warning could avoid reasoning problems in case those classes could not share instances.
- No naming convention is used in the identifiers of the ontology elements. In this case the maintainability, the accessibility and the clarity of the ontology could be improve.
- A cycle between two classes in the hierarchy is included in the ontology. Detecting this situation could avoid modelling and reasoning problems.
- And many other problems described in [the catalogue](#).

Have a look at OOPS! results for the [DBpedia 3.8](#) and [AKT Reference Ontology \(Portal Ontology\)](#) ontologies.

Please, help us making OOPS! better. **Feedback** is more than welcome and you can also [suggest new pitfalls!](#)

Want to help?

- [Suggest new pitfalls](#)
- [Provide feedback](#)

Documentation:

- [Pitfall catalogue](#)
- [User guide](#)
- [Technical report](#)

Related papers:

- [EKAW 2012](#)
- [ESWC 2012 Demo](#)
- [Ontoqual 2010](#)
- [CAEPIA 2009](#)

Web services:

- [RESTful Web Service](#)

Ontology input area

Brief description

Suggestions & feedback



# OOPS! - How it works (i)



**OOPS! (Ontology Pitfall Scanner!)** helps you to detect some of the most common pitfalls appearing when developing ontologies. To try it, enter a URI or paste an OWL document into the text field above. A list of pitfalls and the elements of your ontology where they appear will be displayed.

Scanner by URI:

Example: [http://data.semanticweb.org/ns/swc/swc\\_2009-05-09.rdf](http://data.semanticweb.org/ns/swc/swc_2009-05-09.rdf)

If you just include the RDF code here, the following Pitfalls will not be checked: P36. URI contains file extension, P37. Ontology not available, P40. Namespace hijacking

Scanner by direct input:

☒ Uncheck this checkbox if you don't want us to keep a copy of your ontology.

[Go to advanced evaluation](#)

## News!

Now you can **integrate OOPS!** pitfall detection **with your own developments and tools** simply by invoking the **OOPS! RESTful Web Service**.

## Detecting common pitfalls in ontologies

Modelling ontologies has become one of the main topics of research within ontological engineering because of the difficulties it involves. Developers must tackle a wide range of difficulties and handicaps when modelling ontologies that can imply the appearance of anomalies or errors in ontologies. Therefore, it is important to evaluate the ontologies in order to detect those potential problems.

In this sense, OOPS! helps you to detect some of the most common pitfalls appearing within ontology developments. For example, OOPS! warns you when:

- The domain or range of a relationship is defined as the intersection of two or more classes. This warning could avoid reasoning problems in case those classes could not share instances.
- No naming convention is used in the identifiers of the ontology elements. In this case the maintainability, the accessibility and the clarity of the ontology could be improve.
- A cycle between two classes in the hierarchy is included in the ontology. Detecting this situation could avoid modelling and reasoning problems.
- And many other problems described in [the catalogue](#).

Have a look at OOPS! results for the [DBpedia 3.8](#) and [AKT Reference Ontology \(Portal Ontology\)](#) ontologies.

Please, help us making OOPS! better. **Feedback** is more than welcome and you can also **suggest new pitfalls!**

Want to help?

- [Suggest new pitfalls](#)
- [Provide feedback](#)

Documentation:

- [Pitfall catalogue](#)
- [User guide](#)
- [Technical report](#)

Related papers:

- [EKAW 2012](#)
- [ESWC 2012 Demo](#)
- [Ontoqual 2010](#)
- [CAEPIA 2009](#)

Web services:

- [RESTful Web Service](#)

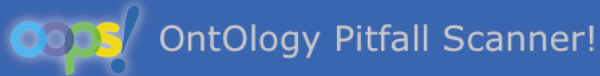
Ontology input area

Brief description

Suggestions & feedback

Documentation

# OOPS! - How it works (i)



**OOPS! (Ontology Pitfall Scanner!)** helps you to detect some of the most common pitfalls appearing when developing ontologies. To try it, enter a URI or paste an OWL document into the text field above. A list of pitfalls and the elements of your ontology where they appear will be displayed.

Scanner by URI:  Scanner by URI

Example: [http://data.semanticweb.org/ns/swc/swc\\_2009-05-09.rdf](http://data.semanticweb.org/ns/swc/swc_2009-05-09.rdf)

If you just include the RDF code here, the following Pitfalls will not be checked: P36. URI contains file extension, P37. Ontology not available, P40. Namespace hijacking

Scanner by direct input:  Scanner by RDF

☒ Uncheck this checkbox if you don't want us to keep a copy of your ontology.

[Go to advanced evaluation](#)

## News!

Now you can **integrate OOPS!** pitfall detection **with your own developments and tools** simply by invoking the **OOPS! RESTful Web Service**.

## Detecting common pitfalls in ontologies

Modelling ontologies has become one of the main topics of research within ontological engineering because of the difficulties it involves. Developers must tackle a wide range of difficulties and handicaps when modelling ontologies that can imply the appearance of anomalies or errors in ontologies. Therefore, it is important to evaluate the ontologies in order to detect those potential problems.

In this sense, OOPS! helps you to detect some of the most common pitfalls appearing within ontology developments. For example, OOPS! warns you when:

- The domain or range of a relationship is defined as the intersection of two or more classes. This warning could avoid reasoning problems in case those classes could not share instances.
- No naming convention is used in the identifiers of the ontology elements. In this case the maintainability, the accessibility and the clarity of the ontology could be improve.
- A cycle between two classes in the hierarchy is included in the ontology. Detecting this situation could avoid modelling and reasoning problems.
- And many other problems described in [the catalogue](#).

Have a look at OOPS! results for the [DBpedia 3.8](#) and [AKT Reference Ontology \(Portal Ontology\)](#) ontologies.

Please, help us making OOPS! better. **Feedback** is more than welcome and you can also **suggest new pitfalls!**

Want to help?

- [Suggest new pitfalls](#)
- [Provide feedback](#)

Documentation:

- [Pitfall catalogue](#)
- [User guide](#)
- [Technical report](#)

Related papers:

- [EKAW 2012](#)
- [ESWC 2012 Demo](#)
- [Ontoqual 2010](#)
- [CAEPIA 2009](#)

Web services:

- [RESTful Web Service](#)

Ontology input area

Brief description

Suggestions & feedback

Documentation

Related papers

# OOPS! - How it works (i)



## OntOlogy Pitfall Scanner!

**OOPS! (OntOlogy Pitfall Scanner!)** helps you to detect some of the most common pitfalls appearing when developing ontologies.

To try it, enter a URI or paste an OWL document into the text field above. A list of pitfalls and the elements of your ontology where they appear will be displayed.

Scanner by URI:

Example: [http://data.semanticweb.org/ns/swc/swc\\_2009-05-09.rdf](http://data.semanticweb.org/ns/swc/swc_2009-05-09.rdf)

Scanner by URI

If you just include the RDF code here, the following Pitfalls will not be checked: P36. URI contains file extension, P37. Ontology not available, P40. Namespace hijacking

Scanner by direct input:

Scanner by RDF

☒ Uncheck this checkbox if you don't want us to keep a copy of your ontology.

[Go to advanced evaluation](#)

### News!

Now you can **integrate OOPS!** pitfall detection **with your own developments and tools** simply by invoking the **OOPS! RESTful Web Service**.

### Detecting common pitfalls in ontologies

Modelling ontologies has become one of the main topics of research within ontological engineering because of the difficulties it involves. Developers must tackle a wide range of difficulties and handicaps when modelling ontologies that can imply the appearance of anomalies or errors in ontologies. Therefore, it is important to evaluate the ontologies in order to detect those potential problems.

In this sense, OOPS! helps you to detect some of the most common pitfalls appearing within ontology developments. For example, OOPS! warns you when:

- The domain or range of a relationship is defined as the intersection of two or more classes. This warning could avoid reasoning problems in case those classes could not share instances.
- No naming convention is used in the identifiers of the ontology elements. In this case the maintainability, the accessibility and the clarity of the ontology could be improve.
- A cycle between two classes in the hierarchy is included in the ontology. Detecting this situation could avoid modelling and reasoning problems.
- And many other problems described in [the catalogue](#).

Have a look at OOPS! results for the [DBpedia 3.8](#) and [AKT Reference Ontology \(Portal Ontology\)](#) ontologies.

Please, help us making OOPS! better. **Feedback** is more than welcome and you can also **suggest new pitfalls!**

Want to help?

- [Suggest new pitfalls](#)
- [Provide feedback](#)

Documentation:

- [Pitfall catalogue](#)
- [User guide](#)
- [Technical report](#)

Related papers:

- [EKAW 2012](#)
- [ESWC 2012 Demo](#)
- [Ontoqual 2010](#)
- [CAEPIA 2009](#)

Web services:

- [RESTful Web Service](#)

Suggestions  
& feedback

Documentation

Related papers

Web Service

Ontology  
input area

Brief  
description

## Evaluation results

[Expand All] | [Collapse All]

### Results for P04: Creating unconnected ontology elements.

11 cases | Minor 🟡

### Results for P05: Defining wrong inverse relationships.

2 cases | Critical 🔴

Two relationships are defined as inverse relations when they are not necessarily. For example, something is sold or something is bought; in this case, the relationships "isSoldIn" and "isBoughtIn" are not inverse.

- This pitfall appears in the following elements:

- > <http://data.semanticweb.org/ns/swc/ontology#relatedToEvent> may not be inverse of <http://data.semanticweb.org/ns/swc/ontology#hasRelatedDocument>
- > <http://data.semanticweb.org/ns/swc/ontology#hasRelatedDocument> may not be inverse of <http://data.semanticweb.org/ns/swc/ontology#relatedToEvent>

### Results for P08: Missing annotations.

156 cases | Minor 🟡

### Results for P11: Missing domain or range in properties.

83 cases | Important 🟠

### Results for P12: Missing equivalent properties.

8 cases | Important 🟠

When an ontology is imported into another, classes that are duplicated in both ontologies are normally defined as equivalent classes. However, the ontology developer misses the definition of equivalent properties in those cases of duplicated relationships and attributes. For example, the classes "CITY" and "City" in two different ontologies are defined as equivalent classes; however, relationships "hasMember" and "has-Member" in two different ontologies are not defined as equivalent relations.

- The following relations could be defined as equivalent:

- > <http://swrc.ontoware.org/ontology#member>, <http://xmlns.com/foaf/0.1/member>
- > <http://purl.org/dc/terms/isPartOf>, <http://data.semanticweb.org/ns/swc/ontology#isPartOf>

- The following attributes could be defined as equivalent:

- > <http://swrc.ontoware.org/ontology#title>, <http://xmlns.com/foaf/0.1/title>
- > <http://purl.org/dc/elements/1.1/source>, <http://swrc.ontoware.org/ontology#source>
- > <http://www.w3.org/2002/12/cal/ical#location>, <http://swrc.ontoware.org/ontology#location>
- > <http://swrc.ontoware.org/ontology#firstName>, <http://xmlns.com/foaf/0.1/firstName>
- > <http://purl.org/dc/elements/1.1/type>, <http://swrc.ontoware.org/ontology#type>
- > <http://swrc.ontoware.org/ontology#date>, <http://www.w3.org/2002/12/cal/ical#date>

Example generated using the ontology [http://data.semanticweb.org/ns/swc/swc\\_2009-05-09.rdf](http://data.semanticweb.org/ns/swc/swc_2009-05-09.rdf)

## Evaluation results

[Expand All] | [Collapse All]

Pitfall name

### Results for P04: Creating unconnected ontology elements.

11 cases | Minor 🟡

### Results for P05: Defining wrong inverse relationships.

2 cases | Critical 🔴

Two relationships are defined as inverse relations when they are not necessarily. For example, something is sold or something is bought; in this case, the relationships "isSoldIn" and "isBoughtIn" are not inverse.

- This pitfall appears in the following elements:

- > <http://data.semanticweb.org/ns/swc/ontology#relatedToEvent> may not be inverse of <http://data.semanticweb.org/ns/swc/ontology#hasRelatedDocument>
- > <http://data.semanticweb.org/ns/swc/ontology#hasRelatedDocument> may not be inverse of <http://data.semanticweb.org/ns/swc/ontology#relatedToEvent>

### Results for P08: Missing annotations.

156 cases | Minor 🟡

### Results for P11: Missing domain or range in properties.

83 cases | Important 🟠

### Results for P12: Missing equivalent properties.

8 cases | Important 🟠

When an ontology is imported into another, classes that are duplicated in both ontologies are normally defined as equivalent classes. However, the ontology developer misses the definition of equivalent properties in those cases of duplicated relationships and attributes. For example, the classes "CITY" and "City" in two different ontologies are defined as equivalent classes; however, relationships "hasMember" and "has-Member" in two different ontologies are not defined as equivalent relations.

- The following relations could be defined as equivalent:

- > <http://swrc.ontoware.org/ontology#member>, <http://xmlns.com/foaf/0.1/member>
- > <http://purl.org/dc/terms/isPartOf>, <http://data.semanticweb.org/ns/swc/ontology#isPartOf>

- The following attributes could be defined as equivalent:

- > <http://swrc.ontoware.org/ontology#title>, <http://xmlns.com/foaf/0.1/title>
- > <http://purl.org/dc/elements/1.1/source>, <http://swrc.ontoware.org/ontology#source>
- > <http://www.w3.org/2002/12/cal/ical#location>, <http://swrc.ontoware.org/ontology#location>
- > <http://swrc.ontoware.org/ontology#firstName>, <http://xmlns.com/foaf/0.1/firstName>
- > <http://purl.org/dc/elements/1.1/type>, <http://swrc.ontoware.org/ontology#type>
- > <http://swrc.ontoware.org/ontology#date>, <http://www.w3.org/2002/12/cal/ical#date>

Example generated using the ontology [http://data.semanticweb.org/ns/swc/swc\\_2009-05-09.rdf](http://data.semanticweb.org/ns/swc/swc_2009-05-09.rdf)

## Evaluation results

[Expand All] | [Collapse All]

Results for P04: Creating unconnected ontology elements.	11 cases   Minor 🟡
Results for P05: Defining wrong inverse relationships.	2 cases   Critical 🔴
<p>Two relationships are defined as inverse relations when they are not necessarily. For example, something is sold or something is bought; in this case, the relationships "isSoldIn" and "isBoughtIn" are not inverse.</p> <ul style="list-style-type: none"> <li>This pitfall appears in the following elements:           <ul style="list-style-type: none"> <li>&gt; <a href="http://data.semanticweb.org/ns/swc/ontology#relatedToEvent">http://data.semanticweb.org/ns/swc/ontology#relatedToEvent</a> may not be inverse of <a href="http://data.semanticweb.org/ns/swc/ontology#hasRelatedDocument">http://data.semanticweb.org/ns/swc/ontology#hasRelatedDocument</a></li> <li>&gt; <a href="http://data.semanticweb.org/ns/swc/ontology#hasRelatedDocument">http://data.semanticweb.org/ns/swc/ontology#hasRelatedDocument</a> may not be inverse of <a href="http://data.semanticweb.org/ns/swc/ontology#relatedToEvent">http://data.semanticweb.org/ns/swc/ontology#relatedToEvent</a></li> </ul> </li> </ul>	
Results for P08: Missing annotations.	156 cases   Minor 🟡
Results for P11: Missing domain or range in properties.	83 cases   Important 🟠
Results for P12: Missing equivalent properties.	8 cases   Important 🟠
<p>When an ontology is imported into another, classes that are duplicated in both ontologies are normally defined as equivalent classes. However, the ontology developer misses the definition of equivalent properties in those cases of duplicated relationships and attributes. For example, the classes "CITY" and "City" in two different ontologies are defined as equivalent classes; however, relationships "hasMember" and "has-Member" in two different ontologies are not defined as equivalent relations.</p> <ul style="list-style-type: none"> <li>The following relations could be defined as equivalent:           <ul style="list-style-type: none"> <li>&gt; <a href="http://swrc.ontoware.org/ontology#member">http://swrc.ontoware.org/ontology#member</a>, <a href="http://xmlns.com/foaf/0.1/member">http://xmlns.com/foaf/0.1/member</a></li> <li>&gt; <a href="http://purl.org/dc/terms/isPartOf">http://purl.org/dc/terms/isPartOf</a>, <a href="http://data.semanticweb.org/ns/swc/ontology#isPartOf">http://data.semanticweb.org/ns/swc/ontology#isPartOf</a></li> </ul> </li> <li>The following attributes could be defined as equivalent:           <ul style="list-style-type: none"> <li>&gt; <a href="http://swrc.ontoware.org/ontology#title">http://swrc.ontoware.org/ontology#title</a>, <a href="http://xmlns.com/foaf/0.1/title">http://xmlns.com/foaf/0.1/title</a></li> <li>&gt; <a href="http://purl.org/dc/elements/1.1/source">http://purl.org/dc/elements/1.1/source</a>, <a href="http://swrc.ontoware.org/ontology#source">http://swrc.ontoware.org/ontology#source</a></li> <li>&gt; <a href="http://www.w3.org/2002/12/cal/ical#location">http://www.w3.org/2002/12/cal/ical#location</a>, <a href="http://swrc.ontoware.org/ontology#location">http://swrc.ontoware.org/ontology#location</a></li> <li>&gt; <a href="http://swrc.ontoware.org/ontology#firstName">http://swrc.ontoware.org/ontology#firstName</a>, <a href="http://xmlns.com/foaf/0.1/firstName">http://xmlns.com/foaf/0.1/firstName</a></li> <li>&gt; <a href="http://purl.org/dc/elements/1.1/type">http://purl.org/dc/elements/1.1/type</a>, <a href="http://swrc.ontoware.org/ontology#type">http://swrc.ontoware.org/ontology#type</a></li> <li>&gt; <a href="http://swrc.ontoware.org/ontology#date">http://swrc.ontoware.org/ontology#date</a>, <a href="http://www.w3.org/2002/12/cal/ical#date">http://www.w3.org/2002/12/cal/ical#date</a></li> </ul> </li> </ul>	

Example generated using the ontology [http://data.semanticweb.org/ns/swc/swc\\_2009-05-09.rdf](http://data.semanticweb.org/ns/swc/swc_2009-05-09.rdf)



# OOPS! - How it works (ii)

## Evaluation results

[Expand All] | [Collapse All]

Pitfall name	Pitfall frequency	Importance Level
Results for P04: Creating unconnected ontology elements.	11 cases   Minor	Minor
Results for P05: Defining wrong inverse relationships.	2 cases   Critical	Critical
Two relationships are defined as inverse relations when they are not necessarily. For example, something is sold or something is bought; in this case, the relationships "isSoldIn" and "isBoughtIn" are not inverse.		
• This pitfall appears in the following elements:		
> <a href="http://data.semanticweb.org/ns/swc/ontology#relatedToEvent">http://data.semanticweb.org/ns/swc/ontology#relatedToEvent</a> may not be inverse of <a href="http://data.semanticweb.org/ns/swc/ontology#hasRelatedDocument">http://data.semanticweb.org/ns/swc/ontology#hasRelatedDocument</a>		
> <a href="http://data.semanticweb.org/ns/swc/ontology#hasRelatedDocument">http://data.semanticweb.org/ns/swc/ontology#hasRelatedDocument</a> may not be inverse of <a href="http://data.semanticweb.org/ns/swc/ontology#relatedToEvent">http://data.semanticweb.org/ns/swc/ontology#relatedToEvent</a>		
Results for P08: Missing annotations.	156 cases   Minor	Minor
Results for P11: Missing domain or range in properties.	83 cases   Important	Important
Results for P12: Missing equivalent properties.	8 cases   Important	Important
When an ontology is imported into another, classes that are duplicated in both ontologies are normally defined as equivalent classes. However, the ontology developer misses the definition of equivalent properties in those cases of duplicated relationships and attributes. For example, the classes "CITY" and "City" in two different ontologies are defined as equivalent classes; however, relationships "hasMember" and "has-Member" in two different ontologies are not defined as equivalent relations.		
• The following relations could be defined as equivalent:		
> <a href="http://swrc.ontoware.org/ontology#member">http://swrc.ontoware.org/ontology#member</a> , <a href="http://xmlns.com/foaf/0.1/member">http://xmlns.com/foaf/0.1/member</a>		
> <a href="http://purl.org/dc/terms/isPartOf">http://purl.org/dc/terms/isPartOf</a> , <a href="http://data.semanticweb.org/ns/swc/ontology#isPartOf">http://data.semanticweb.org/ns/swc/ontology#isPartOf</a>		
• The following attributes could be defined as equivalent:		
> <a href="http://swrc.ontoware.org/ontology#title">http://swrc.ontoware.org/ontology#title</a> , <a href="http://xmlns.com/foaf/0.1/title">http://xmlns.com/foaf/0.1/title</a>		
> <a href="http://purl.org/dc/elements/1.1/source">http://purl.org/dc/elements/1.1/source</a> , <a href="http://swrc.ontoware.org/ontology#source">http://swrc.ontoware.org/ontology#source</a>		
> <a href="http://www.w3.org/2002/12/cal/ical#location">http://www.w3.org/2002/12/cal/ical#location</a> , <a href="http://swrc.ontoware.org/ontology#location">http://swrc.ontoware.org/ontology#location</a>		
> <a href="http://swrc.ontoware.org/ontology#firstName">http://swrc.ontoware.org/ontology#firstName</a> , <a href="http://xmlns.com/foaf/0.1/firstName">http://xmlns.com/foaf/0.1/firstName</a>		
> <a href="http://purl.org/dc/elements/1.1/type">http://purl.org/dc/elements/1.1/type</a> , <a href="http://swrc.ontoware.org/ontology#type">http://swrc.ontoware.org/ontology#type</a>		
> <a href="http://swrc.ontoware.org/ontology#date">http://swrc.ontoware.org/ontology#date</a> , <a href="http://www.w3.org/2002/12/cal/ical#date">http://www.w3.org/2002/12/cal/ical#date</a>		

Example generated using the ontology [http://data.semanticweb.org/ns/swc/swc\\_2009-05-09.rdf](http://data.semanticweb.org/ns/swc/swc_2009-05-09.rdf)

# OOPS! - How it works (ii)

## Evaluation results

[Expand All] | [Collapse All]

<b>Results for P04: Creating unconnected ontology elements.</b>	11 cases   Minor 🟡	Importance Level
<b>Results for P05: Defining wrong inverse relationships.</b>	2 cases   Critical 🔴	
<p>Two relationships are defined as inverse relations when they are not necessarily. For example, something is sold or something is bought; in this case, the relationships "isSoldIn" and "isBoughtIn" are not inverse.</p> <ul style="list-style-type: none"><li>• This pitfall appears in the following elements:<ul style="list-style-type: none"><li>&gt; <a href="http://data.semanticweb.org/ns/swc/ontology#relatedToEvent">http://data.semanticweb.org/ns/swc/ontology#relatedToEvent</a> may not be inverse of <a href="http://data.semanticweb.org/ns/swc/ontology#hasRelatedDocument">http://data.semanticweb.org/ns/swc/ontology#hasRelatedDocument</a></li><li>&gt; <a href="http://data.semanticweb.org/ns/swc/ontology#hasRelatedDocument">http://data.semanticweb.org/ns/swc/ontology#hasRelatedDocument</a> may not be inverse of <a href="http://data.semanticweb.org/ns/swc/ontology#relatedToEvent">http://data.semanticweb.org/ns/swc/ontology#relatedToEvent</a></li></ul></li></ul>		Pitfall description
<b>Results for P08: Missing annotations.</b>	156 cases   Minor 🟡	
<b>Results for P11: Missing domain or range in properties.</b>	83 cases   Important 🟠	
<b>Results for P12: Missing equivalent properties.</b>	8 cases   Important 🟠	
<p>When an ontology is imported into another, classes that are duplicated in both ontologies are normally defined as equivalent classes. However, the ontology developer misses the definition of equivalent properties in those cases of duplicated relationships and attributes. For example, the classes "CITY" and "City" in two different ontologies are defined as equivalent classes; however, relationships "hasMember" and "has-Member" in two different ontologies are not defined as equivalent relations.</p> <ul style="list-style-type: none"><li>• The following relations could be defined as equivalent:<ul style="list-style-type: none"><li>&gt; <a href="http://swrc.ontoware.org/ontology#member">http://swrc.ontoware.org/ontology#member</a>, <a href="http://xmlns.com/foaf/0.1/member">http://xmlns.com/foaf/0.1/member</a></li><li>&gt; <a href="http://purl.org/dc/terms/isPartOf">http://purl.org/dc/terms/isPartOf</a>, <a href="http://data.semanticweb.org/ns/swc/ontology#isPartOf">http://data.semanticweb.org/ns/swc/ontology#isPartOf</a></li></ul></li><li>• The following attributes could be defined as equivalent:<ul style="list-style-type: none"><li>&gt; <a href="http://swrc.ontoware.org/ontology#title">http://swrc.ontoware.org/ontology#title</a>, <a href="http://xmlns.com/foaf/0.1/title">http://xmlns.com/foaf/0.1/title</a></li><li>&gt; <a href="http://purl.org/dc/elements/1.1/source">http://purl.org/dc/elements/1.1/source</a>, <a href="http://swrc.ontoware.org/ontology#source">http://swrc.ontoware.org/ontology#source</a></li><li>&gt; <a href="http://www.w3.org/2002/12/cal/ical#location">http://www.w3.org/2002/12/cal/ical#location</a>, <a href="http://swrc.ontoware.org/ontology#location">http://swrc.ontoware.org/ontology#location</a></li><li>&gt; <a href="http://swrc.ontoware.org/ontology#firstName">http://swrc.ontoware.org/ontology#firstName</a>, <a href="http://xmlns.com/foaf/0.1/firstName">http://xmlns.com/foaf/0.1/firstName</a></li><li>&gt; <a href="http://purl.org/dc/elements/1.1/type">http://purl.org/dc/elements/1.1/type</a>, <a href="http://swrc.ontoware.org/ontology#type">http://swrc.ontoware.org/ontology#type</a></li><li>&gt; <a href="http://swrc.ontoware.org/ontology#date">http://swrc.ontoware.org/ontology#date</a>, <a href="http://www.w3.org/2002/12/cal/ical#date">http://www.w3.org/2002/12/cal/ical#date</a></li></ul></li></ul>		

Example generated using the ontology [http://data.semanticweb.org/ns/swc/swc\\_2009-05-09.rdf](http://data.semanticweb.org/ns/swc/swc_2009-05-09.rdf)



# OOPS! - How it works (ii)

## Evaluation results

[Expand All] | [Collapse All]

Pitfall name	Pitfall frequency	Importance Level	Pitfall description	Ontology elements affected
Results for P04: Creating unconnected ontology elements.	11 cases   Minor	Minor		
Results for P05: Defining wrong inverse relationships.	2 cases   Critical	Critical	<p>Two relationships are defined as inverse relations when they are not necessarily. For example, something is sold or something is bought; in this case, the relationships "isSoldIn" and "isBoughtIn" are not inverse.</p> <ul style="list-style-type: none"><li>This pitfall appears in the following elements:</li><li>&gt; <a href="http://data.semanticweb.org/ns/swc/ontology#relatedToEvent">http://data.semanticweb.org/ns/swc/ontology#relatedToEvent</a> may not be inverse of <a href="http://data.semanticweb.org/ns/swc/ontology#hasRelatedDocument">http://data.semanticweb.org/ns/swc/ontology#hasRelatedDocument</a></li><li>&gt; <a href="http://data.semanticweb.org/ns/swc/ontology#hasRelatedDocument">http://data.semanticweb.org/ns/swc/ontology#hasRelatedDocument</a> may not be inverse of <a href="http://data.semanticweb.org/ns/swc/ontology#relatedToEvent">http://data.semanticweb.org/ns/swc/ontology#relatedToEvent</a></li></ul>	
Results for P08: Missing annotations.	156 cases   Minor	Minor		
Results for P11: Missing domain or range in properties.	83 cases   Important	Important		
Results for P12: Missing equivalent properties.	8 cases   Important	Important	<p>When an ontology is imported into another, classes that are duplicated in both ontologies are normally defined as equivalent classes. However, the ontology developer misses the definition of equivalent properties in those cases of duplicated relationships and attributes. For example, the classes "CITY" and "City" in two different ontologies are defined as equivalent classes; however, relationships "hasMember" and "has-Member" in two different ontologies are not defined as equivalent relations.</p> <ul style="list-style-type: none"><li>The following relations could be defined as equivalent:</li><li>&gt; <a href="http://swrc.ontoware.org/ontology#member">http://swrc.ontoware.org/ontology#member</a>, <a href="http://xmlns.com/foaf/0.1/member">http://xmlns.com/foaf/0.1/member</a></li><li>&gt; <a href="http://purl.org/dc/terms/isPartOf">http://purl.org/dc/terms/isPartOf</a>, <a href="http://data.semanticweb.org/ns/swc/ontology#isPartOf">http://data.semanticweb.org/ns/swc/ontology#isPartOf</a></li><li>The following attributes could be defined as equivalent:</li><li>&gt; <a href="http://swrc.ontoware.org/ontology#title">http://swrc.ontoware.org/ontology#title</a>, <a href="http://xmlns.com/foaf/0.1/title">http://xmlns.com/foaf/0.1/title</a></li><li>&gt; <a href="http://purl.org/dc/elements/1.1/source">http://purl.org/dc/elements/1.1/source</a>, <a href="http://swrc.ontoware.org/ontology#source">http://swrc.ontoware.org/ontology#source</a></li><li>&gt; <a href="http://www.w3.org/2002/12/cal/ical#location">http://www.w3.org/2002/12/cal/ical#location</a>, <a href="http://swrc.ontoware.org/ontology#location">http://swrc.ontoware.org/ontology#location</a></li><li>&gt; <a href="http://swrc.ontoware.org/ontology#firstName">http://swrc.ontoware.org/ontology#firstName</a>, <a href="http://xmlns.com/foaf/0.1/firstName">http://xmlns.com/foaf/0.1/firstName</a></li><li>&gt; <a href="http://purl.org/dc/elements/1.1/type">http://purl.org/dc/elements/1.1/type</a>, <a href="http://swrc.ontoware.org/ontology#type">http://swrc.ontoware.org/ontology#type</a></li><li>&gt; <a href="http://swrc.ontoware.org/ontology#date">http://swrc.ontoware.org/ontology#date</a>, <a href="http://www.w3.org/2002/12/cal/ical#date">http://www.w3.org/2002/12/cal/ical#date</a></li></ul>	


Example generated using the ontology [http://data.semanticweb.org/ns/swc/swc\\_2009-05-09.rdf](http://data.semanticweb.org/ns/swc/swc_2009-05-09.rdf)

- Introduction
- Pitfall Catalogue
- OOPS! (OntOlogy Pitfall Scanner!)
- **Conclusions and Future Work**

# Conclusions and Future Work (i)

Conclusions	
Catalogue	<ul style="list-style-type: none"><li>• Currently <b>40</b> pitfalls including other authors' work</li><li>• <b>Maintained</b> and open to users/experts/practitioners point of view (<b>collaborative</b>)</li></ul>
<b>OOPS!</b> OntOlogy Pitfall Scanner!	<ul style="list-style-type: none"><li>• It is <b>freely available</b> to users on the Web: <a href="http://www.oeg-upm.net/oops">http://www.oeg-upm.net/oops</a><ul style="list-style-type: none"><li>○ is fully <b>independent</b> of any ontology <b>development environment</b> .</li><li>○ works with <b>main web browsers</b> (Firefox, Chrome, Safari and IE).</li><li>○ does <b>not</b> involve <b>installation</b> process.</li><li>○ RESTFul <b>web service</b> available</li></ul></li><li>• Everyone can test it, provide <b>feedback</b>, <b>suggest new pitfalls</b> to be included in the catalogue and implemented into the tool.<ul style="list-style-type: none"><li>○ easy to use<ul style="list-style-type: none"><li>• feedback from a number of users by emails and feedback form</li></ul></li><li>○ broadly used<ul style="list-style-type: none"><li>• <b>&gt;2000</b> executions from 48 countries</li><li>• <b>&gt;600</b> different ontologies</li></ul></li></ul></li><li>• <b>Importance</b> indicators</li><li>• <b>Linked Data</b> requirements considered</li><li>• Integrated within third-party software: OntoHub (<a href="https://ontohub.org/">https://ontohub.org/</a>), SmartCity catalogue (<a href="http://smartcity.linkeddata.es/">http://smartcity.linkeddata.es/</a>), Widoco (<a href="https://github.com/dgarijo/Widoco/">https://github.com/dgarijo/Widoco/</a>)</li></ul>

# Conclusions and Future Work (ii)

**Ontohub** BETA

RepositoriesOntologiesCategoriesLogicsMappingsMore ▾Help

Sign in/up

 **Bioportal**

OverviewOntologiesFile browserHistorySettings

 **Protein-Protein Interaction Ontology** OWL

Ontology defined in the file [/bioportal/PPIO.obo](#)  
<http://ontohub.org/bioportal/PPIO>

A structured controlled vocabulary for the annotation of experiments concerned with protein-protein interactions. Developed by the HUPO Proteomics Standards Initiative.

ContentCommentsMetadataVersionsGraphsMappings

AnnotationProperties **10**Classes **2432**Individuals **1**ObjectProperties **6**Sentences **2996**

12345...Next ›Last »

25 ▾ per page

MS 1002518

MS 1002517

MS 1002516

MS 1002515

https://ontohub.org/

Evaluate ▾

design with OOPS

# Conclusions and Future Work (ii)

## Bioportal

[Overview](#) [Ontologies](#) [File browser](#) [History](#) [Settings](#)

### Protein-Protein Interaction Ontology OWL

Ontology defined in the file [/bioportal/PPIO.obo](#)  
<http://ontohub.org/bioportal/PPIO>

A structured controlled vocabulary for the annotation of experiments concerned with protein-protein interactions. Developed by the HUPO Proteome

[Content](#) [Comments](#) [Metadata](#) [Versions](#) [Graphs](#) [Mappings](#)

[AnnotationProperties](#) 10 [Classes](#) 2432 [Individuals](#) 1 [ObjectProperties](#) 6 [Sentences](#) 2996

[1](#) [2](#) [3](#) [4](#) [5](#) [...](#) [Next »](#) [Last »](#)

[25](#) per page

MS 1002518

MS 1002517

MS 1002516

MS 1002515

<https://ontohub.org/>




trade

<http://smartcity.linkeddata.es/>

**Title** trade  
**URI** <http://personal.us.es/aparedes/Trade.owl>  
**Description** This ontology defines the classes, properties and individuals that make up the commercial management specially focused to purchase orders, in a company dedicated primarily to trade in electrical, energy and environmental products.  
**Languages** English  
**Ontology languages** OWL  
**Ontology format** RDF/XML  
**Issued** 2012-2-28  
**Version** 2.0

## Evaluation results

The following evaluation results have been generated by the RESTful web service provided by OOPS! (Ontology Pitfall Scanner!).

 It is obvious that not all the pitfalls are equally important; their impact in the ontology will depend on multiple factors. For this reason, each pitfall has an importance level attached indicating how important it is. We have identified three levels:

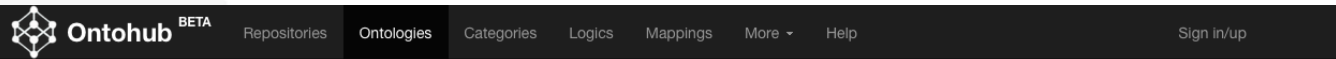
**Critical** It is crucial to correct the pitfall. Otherwise, it could affect the ontology consistency, reasoning, applicability, etc.  
**Important** Though not critical for ontology function, it is important to correct this type of pitfall.  
**Minor** It is not really a problem, but by correcting it we will make the ontology nicer.

<b>P04. Creating unconnected ontology elements</b>	2 cases detected. <small>Minor</small>
Ontology elements (classes, relationships or attributes) are created with no relation to the rest of the ontology. An example of this type of pitfall is to create the relationship "memberOfTeam" and to miss the class representing teams; thus, the relationship created is isolated in the ontology.  This pitfall affects to the following ontology elements: <ul style="list-style-type: none"><li><a href="http://localhost:8080/trade.owl#Trade_Handling">http://localhost:8080/trade.owl#Trade_Handling</a></li><li><a href="http://swrl.stanford.edu/ontologies/built-ins/3.3/temporal.owl#Entity">http://swrl.stanford.edu/ontologies/built-ins/3.3/temporal.owl#Entity</a></li></ul>	
<b>P08. Missing annotations</b>	259 cases detected. <small>Minor</small>
<b>P11. Missing domain or range in properties</b>	27 cases detected. <small>Important</small>
<b>P13. Missing inverse relationships</b>	108 cases detected. <small>Minor</small>
<b>P22. Using different naming criteria in the ontology</b>	ontology <small>Minor</small>
<b>P27. Defining wrong equivalent relationships</b>	1 case detected. <small>Critical</small>
<b>P36. URI contains file extension</b>	ontology <small>Minor</small>

### References:

- [1] Gómez-Pérez, A. Ontology Evaluation. Handbook on Ontologies. S. Staab and R. Studer Editors. Springer. International Handbooks on Information Systems. Pp. 251 – 274. 2004.
- [2] Noy, N.F., McGuinness, D. L. Ontology development 101: A guide to creating your first ontology. Technical Report SMI-2001-0880, Stanford Medical Informatics. 2001.
- [3] Rector, A., Drummond, N., Horridge, M., Rogers, J., Knublauch, H., Stevens, R., Wang, H., Wroe, C. "Owl pizzas: Practical experience of teaching owl-di: Common errors and common patterns". In Proc. of EKAW 2004, pp: 63 – 81. Springer. 2004.
- [4] Hogan, A., Harth, A., Passant, A., Decker, S., Polleres, A. Weaving the Pedantic Web. Linked Data on the Web Workshop LDOW2010 at WWW2010 (2010).
- [5] Archer, P., Goedertier, S., and Loutas, N. D7.1.3 – Study on persistent URIs, with identification of best practices and recommendations on the topic for the MSs and the EC. Deliverable. December 17, 2012.
- [6] Heath, T., Bizer, C.: Linked data: Evolving the Web into a global data space (1st edition). Morgan & Claypool (2011).

# Conclusions and Future Work (ii)



Bioportal

Overview Ontologies File browser History Settings

## Protein-Protein Interaction Ontology OWL

Ontology defined in the file [/bioportal/PPIO.obo](http://bioportal/PPIO.obo)  
<http://ontohub.org/bioportal/PPIO>

A structured controlled vocabulary for the annotation of experiments concerned with protein-protein interactions. Developed by the HUPO Proteome

Content Comments Metadata Versions Graphs Mappings

Annotation Properties 10 Classes 2432 Individuals 1 Object Properties 6 Sentences 2996

1 2 3 4 5 ... Next » Last »

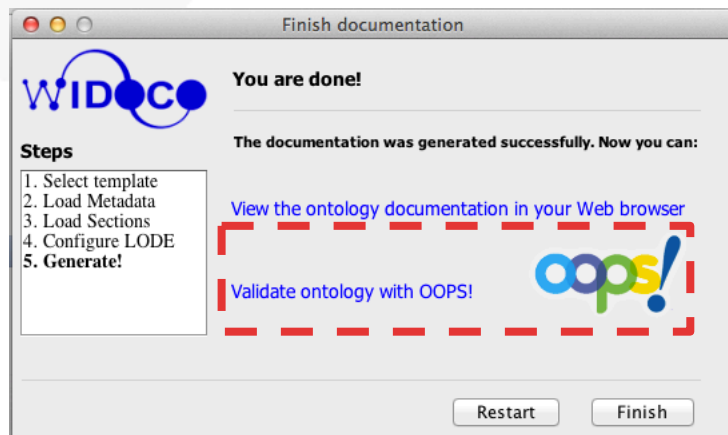
25 per page

MS 1002518

MS 1002517

MS 1002516

MS 1002515



<https://github.com/dgarijo/Widoco/>

<https://ontohub.org/>



trade

<http://smartcity.linkeddata.es/>

Title	trade
URI	<a href="http://personal.us.es/aparedes/Trade.owl">http://personal.us.es/aparedes/Trade.owl</a>
Description	This ontology defines the classes, properties and individuals that make up the commercial management specially focused to purchase orders, in a company dedicated primarily to trade in electrical, energy and environmental products.
Languages	English
Ontology languages	OWL
Ontology format	RDF/XML
Issued	2012-2-28
Version	2.0

### Evaluation results

The following evaluation results have been generated by the RESTful web service provided by OOPS! (Ontology Pitfall Scanner!).

It is obvious that not all the pitfalls are equally important; their impact in the ontology will depend on multiple factors. For this reason, each pitfall has an importance level attached indicating how important it is. We have identified three levels:

Critical	It is crucial to correct the pitfall. Otherwise, it could affect the ontology consistency, reasoning, applicability, etc.
Important	Though not critical for ontology function, it is important to correct this type of pitfall.
Minor	It is not really a problem, but by correcting it we will make the ontology nicer.

P04. Creating unconnected ontology elements	2 cases detected. Minor
Ontology elements (classes, relationships or attributes) are created with no relation to the rest of the ontology. An example of this type of pitfall is to create the relationship "memberOfTeam" and to miss the class representing teams; thus, the relationship created is isolated in the ontology.	
This pitfall affects to the following ontology elements: <ul style="list-style-type: none"><li><a href="http://localhost:8080/trade.owl#Trade_Handling">http://localhost:8080/trade.owl#Trade_Handling</a></li><li><a href="http://swrl.stanford.edu/ontologies/built-ins/3.3/temporal.owl#Entity">http://swrl.stanford.edu/ontologies/built-ins/3.3/temporal.owl#Entity</a></li></ul>	
P08. Missing annotations	259 cases detected. Minor
P11. Missing domain or range in properties	27 cases detected. Important
P13. Missing inverse relationships	108 cases detected. Minor
P22. Using different naming criteria in the ontology	ontology Minor
P27. Defining wrong equivalent relationships	1 case detected. Critical
P36. URI contains file extension	ontology Minor

#### References:

- [1] Gómez-Pérez, A. Ontology Evaluation. Handbook on Ontologies. S. Staab and R. Studer Editors. Springer, International Handbooks on Information Systems. Pp. 251 – 274, 2004.
- [2] Noy, N.F., McGuinness, D. L. Ontology development 101: A guide to creating your first ontology. Technical Report SMI-2001-0880, Stanford Medical Informatics, 2001.
- [3] Rector, A., Drummond, N., Horridge, M., Rogers, J., Knublauch, H., Stevens, R., Wang, H., Wroe, C. "Owl pizzas: Practical experience of teaching owl-di: Common errors and common patterns". In Proc. of EKAW 2004, pp: 63 – 81. Springer, 2004.
- [4] Hogan, A., Harth, A., Passant, A., Decker, S., Polleres, A. Weaving the Pedantic Web. Linked Data on the Web Workshop LDOW2010 at WWW2010 (2010).
- [5] Archer, P., Goedertier, S., and Loutas, N. D7.1.3 – Study on persistent URIs, with identification of best practices and recommendations on the topic for the MSs and the EC. Deliverable, December 17, 2012.
- [6] Heath, T., Bizer, C.: Linked data: Evolving the Web into a global data space (1st edition). Morgan & Claypool (2011).

# Conclusions and Future Work (iii)

## Future Work

<b>Catalogue</b>	<ul style="list-style-type: none"><li>• Continuous maintenance</li><li>• To include <b>guidelines</b> about how <b>to solve</b> each pitfall</li></ul>
<b>OOPS!</b> OntOlogy Pitfall Scanner!	<ul style="list-style-type: none"><li>• To <b>automate</b> the <b>remaining</b> 8 pitfalls</li><li>• To <b>improve</b> the <b>detection methods</b> of some of the already implemented pitfalls (mainly by using linguistic resources)</li><li>• To allow pitfalls <b>definition</b> following a <b>formal language</b>, according with their particular quality criteria</li><li>• To <b>integrate</b> it in <b>ontology editors</b> (E.g: NeOn Toolkit and Protégé)</li><li>• To provide compliance badges (on going work)</li></ul>

## Publications

- M. Poveda-Villalón, A. Gómez-Pérez, M.C. Suárez-Figueroa. **OOPS! (Ontology Pitfall Scanner!): An On-line Tool for Ontology Evaluation**. *International Journal on Semantic Web and Information Systems (IJSWIS) Volumen 10(2) special issue on Web Data Quality*. **To appear**
- M. Poveda-Villalón, M.C. Suárez-Figueroa, A. Gómez-Pérez. **Validating ontologies with OOPS!**. 18<sup>th</sup> International Conference on Knowledge Engineering and Knowledge Management (EKAW2012). 8 - 12 October 2012, Galway, Ireland. ISBN:978-3-642-33875-5

Thanks for your attention!





EUON Workshop, 25<sup>th</sup> September 2014

# **OOPS! (ONTOLOGY PITFALL SCANNER!): A WEB-BASED TOOL FOR ONTOLOGY EVALUATION**

**María Poveda-Villalón, Mari Carmen Suárez-Figueroa and  
Asunción Gómez-Pérez**

Ontology Engineering Group.

Universidad Politécnica de Madrid.

{**mpoveda**, mcsuarez, asun}@fi.upm.es

OOPS! website: <http://www.oeg-upm.net/oops>

Twitter account: **@OOPSoeg**

